3

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/577,967	05/25/2000	Rafiul Ahad	50277-456 [OID-1999-048-0	7503

7590 04/06/2005

Ditthavong & Carlson P C
10507 Braddock Rd
Suite A
Fairfax, VA 22032

EXAMINER	
STEELMAN, MARY J	

ART UNIT	PAPER NUMBER
2191	

DATE MAILED: 04/06/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 09/577,967	**Applicant(s)** AHAD, RAFIUL	
	Examiner Mary J. Steelman	**Art Unit** 2191	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _30 November 2004_.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1-17_ is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) _1-17_ is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) □ Notice of References Cited (PTO-892)
2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) □ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .
5) □ Notice of Informal Patent Application (PTO-152)
6) □ Other: _____.

DETAILED ACTION

1. This Office Action is in response to Remarks and Amendments filed 30 November 2004. Claims 1, 3, 5, and 9-11 are amended. Claims 1-17 are pending.

Claim Rejections - 35 USC § 103

2. Claims 1-17 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent 6,374,256 B1 to Ng et al., in view of US Patent 6,321,261 B1 to Glass.

Per claims 1, 13, 14, 15 and 17, Ng disclosed:

-instantiating a persistent agent…, said persistent agent providing an interface including a routine for persistently storing the object in a persistent object store;

(Col. 6, lines 13-35. "…database related instructions may use (instantiate a persistent agent) database connectivity module (for sessions) DBC to connect object-oriented database runtime module with database information. DBC provides an application programming interface (API) (persistent agent) for programmers to access a database…the JDBC™ database connectivity module is one type of DBC that provides an interface between Java™ applications and a database…It consists of a set of classes (based on a class) and interfaces (providing an interface)…It provides a standard API for tool/database developers and makes it possible to write database applications (routine for persistent object store)." Also, col. 5, lines 10-19, "Classes associated with a database are considered "persistence-capable" classes. The objects representing these persistence-classes can be either persistent objects or transient objects depending how they are instantiated. If an object is a persistent instance of a persistent-capable class, the object is stored in the database and is accessible to many different applications.")

-storing the object in the persistent object store by invoking the routine via the interface provided by the persistent agent.

(Col. 5, lines 13-16, "If an object is a persistent instance of a persistent-capable class, the object is stored in the database and is accessible to many different applications", col. 5, lines 23-25, "In general, object-oriented applications that access persistent objects send requests (invoking a routine)...", col. 10, lines 53-57, "Tool also generates default methods to operate on each field in a class. A 'get' method is created to get a field value (invoke a 'get' method) stored in a persistent object in the tables of a database. Similarly, a 'set' method is used to set a value (invoke a 'set' method) in each field of a persistent object also stored in the database.")

While Ng suggested that agents be instantiated (col. 7, lines 9-10), "Either applet or application generates persistence-capable objects", he failed to disclose "creating a persistent agent based on a name identifying the class". Glass provided more details on naming conventions used when constructing an object.

Glass disclosed (col. 3, line 50-col. 4, line 5), "When vcc utility is run on Store.class, the new class Vstore.class is created. When vcc is executed **with the name of a class**..." (emphasis added), "Thereafter, to construct a remote object or agent (instantiate an agent) of the class (how agent is named: instantiate agent based on a name identifying the class) Store, the following exemplary syntax may be used: Vstore vstore = new Vstore ("dallas:8000/Store1");

(agent is instantiated based on a name identifying the class)…", "By default, the name of the created object is set to a globally unique collection of bytes, but an optional string alias may be assigned. A conventional URL (uniform resource locator) syntax may also be used to refer to the object. For example, the new remote object with alias "Store1" is located at a remote host or IP address of 'dallas' at port number '8000' with the above construction syntax. Note that the construction syntax follows conventional JAVA construction syntax with an enhanced or extended interface that accepts the string address and optional alias…"

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to have modified Ng's invention, to include details regarding the naming of created persistent agents, because Ng recognized the need to improve the techniques for mapping object-oriented application and databases (col. 2, lines 5-10), while Glass recognized the need for supporting communications (col. 1, lines 35-42). To facilitate the storage of a object of a named class by instantiating an agent, based on a name identifying the class, provides a logical, mnemonic relationship / mapping when managing an object oriented persistent store.

Per claim 2:

-modifying the object in the persistent object store based on the persistent agent.

(Ng: Col. 8, lines 52-54, "…stored procedures (based on the persistent agent) can be generated to perform routine operations on objects such as Add, Delete, Modify and Retrieve data entries (objects) in the database (persistent object store).")

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to have modified Ng's invention, to include details regarding the naming of created persistent agents, because Ng recognized the need to improve the techniques for mapping object-oriented application and databases (col. 2, lines 5-10), while Glass recognized the need for supporting communications (col. 1, lines 35-42). To facilitate the storage of a object of a named class by instantiating an agent, based on a name identifying the class, provides a logical, mnemonic relationship / mapping when managing an object oriented persistent store.

Per claim 3:

-instantiating the persistent agent based on a fully qualified name for the class.

(Ng: Col. 5, lines 43-45, "Processor executes instructions (instantiates) associated with applications contained in memory..." and lines 51-53, "Memory includes...a database connectivity (DBC) module..."

Also see response to claim 1 above, "DBC provides an application programming interface (API) (persistent agent) for programmers to access a database..." and "It provides a standard API for tool/database developers and makes it possible to write database applications (persistent object store) using a pure Java^{TM} API (based on a fully qualified name for a JAVA class).")

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to have modified Ng's invention, to include details regarding the naming of created persistent agents, because Ng recognized the need to improve the techniques for mapping object-oriented application and databases (col. 2, lines 5-10), while Glass recognized the need for supporting communications (col. 1, lines 35-42). To facilitate the storage of a object of a named class by instantiating an agent, based on a name identifying the class, provides a logical, mnemonic relationship / mapping when managing an object oriented persistent store.

Per claim 4:

-the persistent object store includes a relational database;

(Ng: Col. 5, lines 20-22, "Object-database server provides a conduit between a relational database connected to database server an object-oriented applications…")

-storing the object in the persistent object store includes the step of storing the object in at least one database table corresponding to the class. (Col. 8, lines 1-3, "…the class-to-database mapping maps (stores) one class in the object-oriented application to one table in the database.")

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to have modified Ng's invention, to include details regarding the naming of created persistent agents, because Ng recognized the need to improve the techniques for mapping object-oriented application and databases (col. 2, lines 5-10), while Glass recognized the need for supporting communications (col. 1, lines 35-42). To facilitate the storage of a object of a named class by instantiating an agent, based on a name identifying the class, provides a logical, mnemonic relationship / mapping when managing an object oriented persistent store.

Per claim 5:

-determining if the at least one database table corresponding to the class has been created; if the at least one database table is determined not to have been created, then creating the at least one database table.

(Ng: Col. 8, lines 12-18, "Tool generates a schema to create (creating the at least one database table) the tables in a database. A schema includes a description of the tables in a database (if created) and their various attributes. A user uses the schema to generate a database. Multiple schemas can be organized as separate entries in a catalogue. The catalogue names each schema with a unique label to distinguish the schemas from each other.")

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to have modified Ng's invention, to include details regarding the naming of created persistent agents, because Ng recognized the need to improve the techniques for mapping object-oriented application and databases (col. 2, lines 5-10), while Glass recognized the need for supporting communications (col. 1, lines 35-42). To facilitate the storage of a object of a named class by instantiating an agent, based on a name identifying the class, provides a logical, mnemonic relationship / mapping when managing an object oriented persistent store.

Per claim 6:

-storing values of at least some of the fields in corresponding columns of the database table.

(Ng: Col. 7, lines 64-65, "Tool creates tables in the database having rows and columns corresponding to the one or more classes." Also, col. 8, lines 5-7, "...each column corresponds to each of the multiple fields.")

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to have modified Ng's invention, to include details regarding the naming of created persistent agents, because Ng recognized the need to improve the techniques for mapping object-oriented application and databases (col. 2, lines 5-10), while Glass recognized the need for supporting communications (col. 1, lines 35-42). To facilitate the storage of a object of a named class by instantiating an agent, based on a name identifying the class, provides a logical, mnemonic relationship / mapping when managing an object oriented persistent store.

Per claim 7:

-designating at least some of the columns as primary key columns based on a list of corresponding field names of the object.

(Ng: Fig. 8 and Col. 9, lines 28-38, "...primary keys can also be used to represent the one-to-many relationships between classes...")

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to have modified Ng's invention, to include details regarding the naming of created persistent agents, because Ng recognized the need to improve the techniques for mapping object-oriented application and databases (col. 2, lines 5-10), while Glass recognized the need for supporting communications (col. 1, lines 35-42). To facilitate the storage of a object of a named class by instantiating an agent, based on a name identifying the class, provides a logical, mnemonic relationship / mapping when managing an object oriented persistent store.

Per claim 8:

-building an index on at least some of the columns based on a list of corresponding field names of the object.

(Ng: Col. 8, lines 19-29, "...specify which fields in an object should be indexed in the database for fast access. The fields specified by the user are logically organized by tool (building an index) in an index group.")

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to have modified Ng's invention, to include details regarding the naming of created persistent agents, because Ng recognized the need to improve the techniques for mapping object-oriented application and databases (col. 2, lines 5-10), while Glass recognized the need for supporting communications (col. 1, lines 35-42). To facilitate the storage of a object of a named class by instantiating an agent, based on a name identifying the class, provides a logical, mnemonic relationship / mapping when managing an object oriented persistent store.

Per claim 9:

-instantiating an other persistent agent based on the other class; storing the other object in the persistent object store based on the other persistent agent.

(Ng: Col. 8, lines 8-11, "Alternatively, more complex class-to-database mappings can be used to map classes to tables such as mapping...multiple classes to a single table.")

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to have modified Ng's invention, to include details regarding the naming of created persistent agents, because Ng recognized the need to improve the techniques for mapping object-oriented application and databases (col. 2, lines 5-10), while Glass recognized the need for supporting communications (col. 1, lines 35-42). To facilitate the storage of a object of a named class by instantiating an agent, based on a name identifying the class, provides a logical, mnemonic relationship / mapping when managing an object oriented persistent store.

Per claim 10:

-the step of instantiating the persistent agent includes the step of creating the persistent agent based on the session.

(Ng: See figs. 1 & 2. Col. 5, lines 43-45, "Processor executes instructions associated with applications contained in memory..." and lines 51-53, "Memory includes...a database connectivity (DBC) module..." and col. 6, lines 13-35, "DBC provides an application programming interface (API) (persistent agent) for programmers to access a database..." and "It provides a standard API for tool/database developers and makes it possible to write database applications using a pure Java^{TM} API...DBC can be used to establish a logical connection (session) with a database..." When a 'session' is desired the DBC is used and provides an interface (a persistent agent), for database access.)

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to have modified Ng's invention, to include details regarding the naming of created persistent agents, because Ng recognized the need to improve the techniques for mapping object-oriented application and databases (col. 2, lines 5-10), while Glass recognized the need for supporting communications (col. 1, lines 35-42). To facilitate the storage of a object of a named class by instantiating an agent, based on a name identifying the class, provides a logical, mnemonic relationship / mapping when managing an object oriented persistent store.

Per claims 11 and 16:

-instantiating a persistent agent based on a name identifying a class, said persistent agent providing an interface including a routine for retrieving a set of objects from the persistent object store;

(Ng: Col. 6, lines 13-35. "...database related instructions may use database connectivity module (for sessions) DBC to connect object-oriented database runtime module with database information. DBC provides an application programming interface (API) (persistent agent) for programmers to access a database...the JDBC^{TM} database connectivity module is one type of DBC that provides an interface between Java^{TM} applications and a database...It consists of a set of classes (based on a class) and interfaces (providing an interface)...It provides a standard API for tool/database developers and makes it possible to write database applications (routine for persistent object retrieval)." Also, col. 5, lines 10-19, "Classes associated with a database are considered "persistence-capable" classes. The objects representing these persistence-classes can be either persistent objects or transient objects depending how they are instantiated. If an object is a persistent instance of a persistent-capable class, the object is stored in the database and is accessible (retrieve a set of objects from the persistent object store) to many different applications.")

-retrieving the set of objects in the persistent object store by invoking the routine via the interface provided by the persistent agent.

(Ng: Col. 4, lines 3-34, "The DBMS also includes standard software development interfaces such as Structured Query Language (SQL) (invoke a routine via the interface provided by the persistent agent), stored procedures, and a variety of concurrence control mechanisms", col. 10, lines 53-60, "Tool also generates default methods to operate on each field in a class…A 'get' method (retrieve) …a 'set' method…to manipulate field values in persistent objects…")

While Ng suggested that agents be created (col. 7, lines 9-10), "Either applet or application generates persistence-capable objects", he failed to disclose "creating a persistent agent based on a name identifying the class". Glass provided more details on naming conventions used when constructing an object.

Glass disclosed (col. 3, line 50-col. 4, line 5), "When vcc utility is run on Store.class, the new class Vstore.class is created. When vcc is executed **with the name of a class**…" (emphasis added), "Thereafter, to construct a remote object or agent (instantiate an agent) of the class (how agent is named: instantiate agent based on a name identifying the class) Store, the following exemplary syntax may be used: Vstore vstore = new Vstore ("dallas:8000/Store1");

(agent is instantiated based on a name identifying the class)...", "By default, the name of the created object is set to a globally unique collection of bytes, but an optional string alias may be assigned. A conventional URL (uniform resource locator) syntax may also be used to refer to the object. For example, the new remote object with alias "Store1" is located at a remote host or IP address of 'dallas' at port number '8000' with the above construction syntax. Note that the construction syntax follows conventional JAVA construction syntax with an enhanced or extended interface that accepts the string address and optional alias..."

"Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to have modified Ng's invention, to include details regarding the naming of created persistent agents, because Ng recognized the need to improve the techniques for mapping object-oriented application and databases (col. 2, lines 5-10), while Glass recognized the need for supporting communications (col. 1, lines 35-42). To facilitate the storage of a object of a named class by instantiating an agent, based on a name identifying the class, provides a logical, mnemonic relationship / mapping when managing an object oriented persistent store.

Per claims 12 and 14:

-retrieving the set of objects in the persistent object store based further on a predicate.

(Ng: Col. 10, lines 40-52, "Tool also generates object query language (OQL) routines for use by objects generated from a particular class...query objects using an object-oriented programming language interface...a query on a particular class will return all objects (retrieving the set of objects) in that class as a connection of objects which match the query criteria (using a predicate).")

"Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention, to have modified Ng's invention, to include details regarding the naming of created persistent agents, because Ng recognized the need to improve the techniques for mapping object-oriented application and databases (col. 2, lines 5-10), while Glass recognized the need for supporting communications (col. 1, lines 35-42). To facilitate the storage of a object of a named class by instantiating an agent, based on a name identifying the class, provides a logical, mnemonic relationship / mapping when managing an object oriented persistent store.

Response to Arguments

3. Applicant has argued, in substance, the following:

Applicant has noted (Remarks, 30 November 2004), page 7, last paragraph through page 8, first paragraph, the combined references fail to disclose "a persistent agent, is instantiated based on a name identifying a class..." Further, page 8, lower half of the page, states that the limitation "instantiating a persistent agent based on a name identifying the class" specifies how the agent is instantiated and not how the agent is named.

Examiner's Response: Note rejection of claim 1 above. The Glass reference disclosed that a class is created based on a name. Col. 3, lines 58-67, "Thereafter, to construct a remote object or agent (how the agent is instantiated, based on a name identifying a class) of the class Store, the following exemplary syntax may be used…

Examiner maintains the rejection of claims 1-17.

Conclusion

4. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

5. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Mary Steelman, whose telephone number is (571) 272-3704. The examiner can normally be reached Monday through Thursday, from 7:00 AM to 5:30 PM If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan

Q. Dam can be reached at (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Mary Steelman
03/30/2005

WEI Y. ZHEN
PRIMARY EXAMINER